A Convolutionally Encoded OSTBC System with SNR-Adaptive Constellations for Low-Latency and Low-Complexity Communications

Mehmet Cagri Ilter[*], Risto Wichman[*], Jyri Hämäläinen[†], and Halim Yanikomeroglu[**]

[*] Department of Signal Processing and Acoustics, Aalto University, Espoo, Finland, [†]Department of Communications and Networking, Aalto University, Espoo, Finland
[**] Department of Systems and Computer Engineering, Carleton University, Ottawa, Ontario, Canada

Motivation

- 5G new radio (NR) aims to facilitate new use cases in wireless communications; low-latency communications.

- Many of the powerful FEC codes deployed in current systems; such as, the turbo and low-density parity-check (LDPC) codes do not perform well when the low-latency requirement does not allow iterative decoding.

- Motivated by this observation, we introduce a convolutionally encoded OSTBC system with SNR-adaptive constellations for low-latency and low-complexity use-cases.

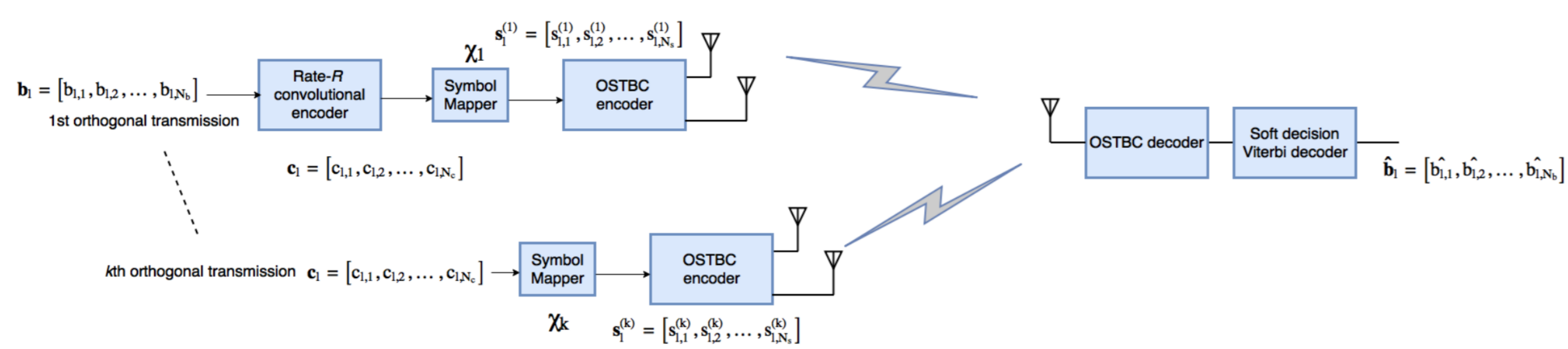## A convolutionally encoded OSTBC system

Figure 1: The block diagram of convolutionally encoded OSTBC system.

We consider a convolutionally encoded OSTBC transmission model which consists of K orthogonal transmission stages as shown in Fig. 1. This model may appear when using coordinated multi-point (CoMP) and relaying where the source and relay link includes error-free transmission. An N_b-length information bit block of frame l, $\mathbf{b}_l$, is first encoded by a rate-R convolutional encoder; then, the encoded bits, $\mathbf{c}_l = [c_{l,1} \cdots c_{l,N_c}]$, are fed into the symbol mapper where a M-ary constellation of choice is applied. Note that a choice of constellation for each transmission stage can differ, while the same encoded bits are considered in each stage. Then, output symbols for kth orthogonal phase, $\mathbf{s}_l^{(k)} = [s_{l,1}^{(k)} \cdots s_{l,N_s}^{(k)}]$, are transmitted after using OSTBC encoder. For instance, the transmitting code-block of the Alamouti coded scheme [1] for kth stage, $\mathbf{S}_l^{(k)}$, can be expressed as

$$\mathbf{S}_l^{(k)} = \begin{bmatrix} s_{l,1}^{(k)} & -s_{l,2}^{(k)*} & \cdots & s_{l,2i-1}^{(k)} & -s_{l,2i}^{(k)*} \\ s_{l,2}^{(k)} & s_{l,1}^{(k)*} & \cdots & s_{l,2i}^{(k)} & s_{l,2i-1}^{(k)*} \end{bmatrix}, \tag{1}$$

where $(\cdot)^*$ denotes the complex conjugate of the corresponding output symbol $i \in \{1, N_s/2\}$. Herein, each column corresponds to different time slot and each row denotes different transmitting antenna.

The channel coefficients during the lth frame, $\mathbf{h}_l = [h_{l,1}, \cdots, h_{l,N_s}]$, are modeled by quasi-static Nakagami-m fading model with a shaping parameter m and an average fading power Ω. We assume independent fading between the K orthogonal transmission stages, with possibly different shaping parameters m_k and link powers Ω_k. The average received SNR at the receiver for kth stage can be explicitly defined as $\bar{\gamma}_k = \Omega_k/N_0$ where Ω_k can be interpreted as the path-loss term.

SNR-adaptive irregular optimized constellations

In order to establish an optimization framework without any constraint on symbol point locations for any given convolutionally encoded scenario, the product-state matrix method [2] is required in the calculation of the upper bound on the BER, P_b. Since this method considers all possible combinations of the encoded and the decoded states rather than simply assuming all zero bits transmitted from the encoder, there are N^2 product-states for a N-state convolutional encoder. By using the product-state matrix technique, upper bound on BER, P_b, can be found as [2]

$$P_b \leq \frac{1}{l} \frac{\partial}{\partial I} \mathbf{1}^T \mathbf{S}_{\mathcal{G}\mathcal{G}} \mathbf{1} + \left(\mathbf{1}^T \mathbf{S}_{\mathcal{G}\mathcal{B}} \right)^T [\mathbf{I} - \mathbf{S}_{\mathcal{B}\mathcal{B}}]^{-1} \mathbf{S}_{\mathcal{B}\mathcal{G}} \mathbf{1} \Big|_{I=1}, \tag{2}$$

where l is the number of information bits per output symbol, $\mathbf{1}$ and $\mathbf{I}$ denote the unity and identity matrices, respectively. The detailed description of the product-state matrix technique for convolutional encoded case can be found in [2].

Once upper bound BER expression is obtained from product-state matrix technique, particle swarm optimization (PSO) algorithm is applied to find SNR-adaptive optimized irregular constellations, $\chi(m, \bar{\gamma})$, which can vary with m and average SNR, $\bar{\gamma} = \Omega/N_0$. The PSO optimizer requires the following parameters before going through the constellation search: the fading parameter m, the average fading power Ω, the modulation order M, the swarm size P, and optimizer parameters $(r_1, r_2, w, N_{\text{iter}})$. The PSO technique has smaller computational load and fewer tuning parameters than other evolutionary algorithms [3]. The used parameters for the constellation search can be found in [4].

How PSO works to find out optimized symbol point locations?

After initializing the positions of P particles', $\mathbf{x}_i^0$, and their velocities, $\mathbf{v}_i^0$, the fitness value of each particle is calculated from (2). Then, the best value of the swarm, g_p^0, which is the one giving the minimum value of P_b, is calculated. The velocities and positions of the particles are updated as follows:

$$x_i^n = x_i^{n-1} + v_i^n \tag{3a}$$

$$v_i^n = v_i^{n-1} + r_1 c_1 \left(g_i^{n-1} - x_i^{n-1} \right) + r_2 c_2 \left(g_p^{n-1} - x_i^{n-1} \right). \tag{3b}$$

Calculations of (2) are carried out by considering these updated values and the ones giving the lowest P_b values are kept as the updated particles. At the end of the Nth iteration, the $x_i^{\bar{N}_{iter}}$ yields to the optimized irregular constellation, $\chi(m, \bar{\gamma})$.
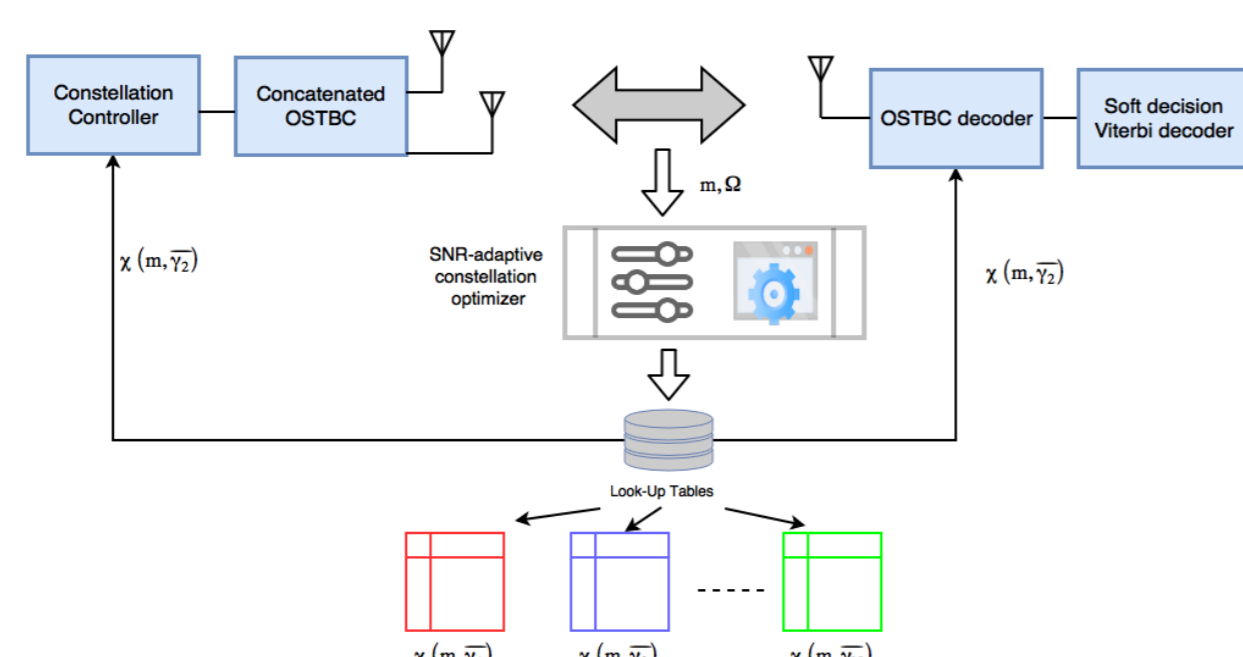

Figure 2: The proposed SNR-adaptive transmission model for the convolutionally encoded OSTBC scheme.

Better error performance and lower decoding latencies

We consider a convolutionally encoded transmission system along with Alamouti scheme in order to validate the error performance improvement due to using the SNR-adaptive optimized irregular constellations. A rate-1/2 convolutional encoder, $[5, 7]_8$ is employed along with Alamouti encoder which consists of two transmit and single receive antenna. We employ Alamouti space-time code, because it is capacity optimal for this set-up. In simulations, different modulation orders and Nakagami shaping parameters, m, are applied to compare the performance improvement in different scenarios. The frame-length, N_b, is set to 120 bits.

The proposed SNR-adaptive transmission model is tested over 16-ary signaling cases along with the different m values. The simulated BER results for SNR-adaptive transmission are compared with Gray-mapped 16-QAM constellations in Fig. 3.a and showing the gain from the SNR-adaptive transmission. Optimized 16-ary constellations for Rayleigh fading are illustrated in Fig. 4 where the symbol point locations tend to gather in four different groups when $\bar{\gamma}$ is increasing.

In order to further demonstrate the superiority of the convolutional encoder over the powerful channel coding techniques, the decoding delay comparison which includes the turbo-TCM encoded scenario with different decoding iterations is given in Fig. 3.b. For simplicity, the window-length of back-search limit in Viterbi decoder, τ, is selected as a quantity of the decoding latency following the approach of [5]. Over the simulations, turbo-TCM encoder given in [6] is employed along with the different numbers of decoding iterations, Q, along with 16-QAM constellation. Although the error performance can be enhanced with increasing number of the decoder iteration, it comes with the expense of higher decoding latency values and convolutionally coded transmissions can be a good option to reach lower decoding latencies compared to turbo-TCM scenarios.
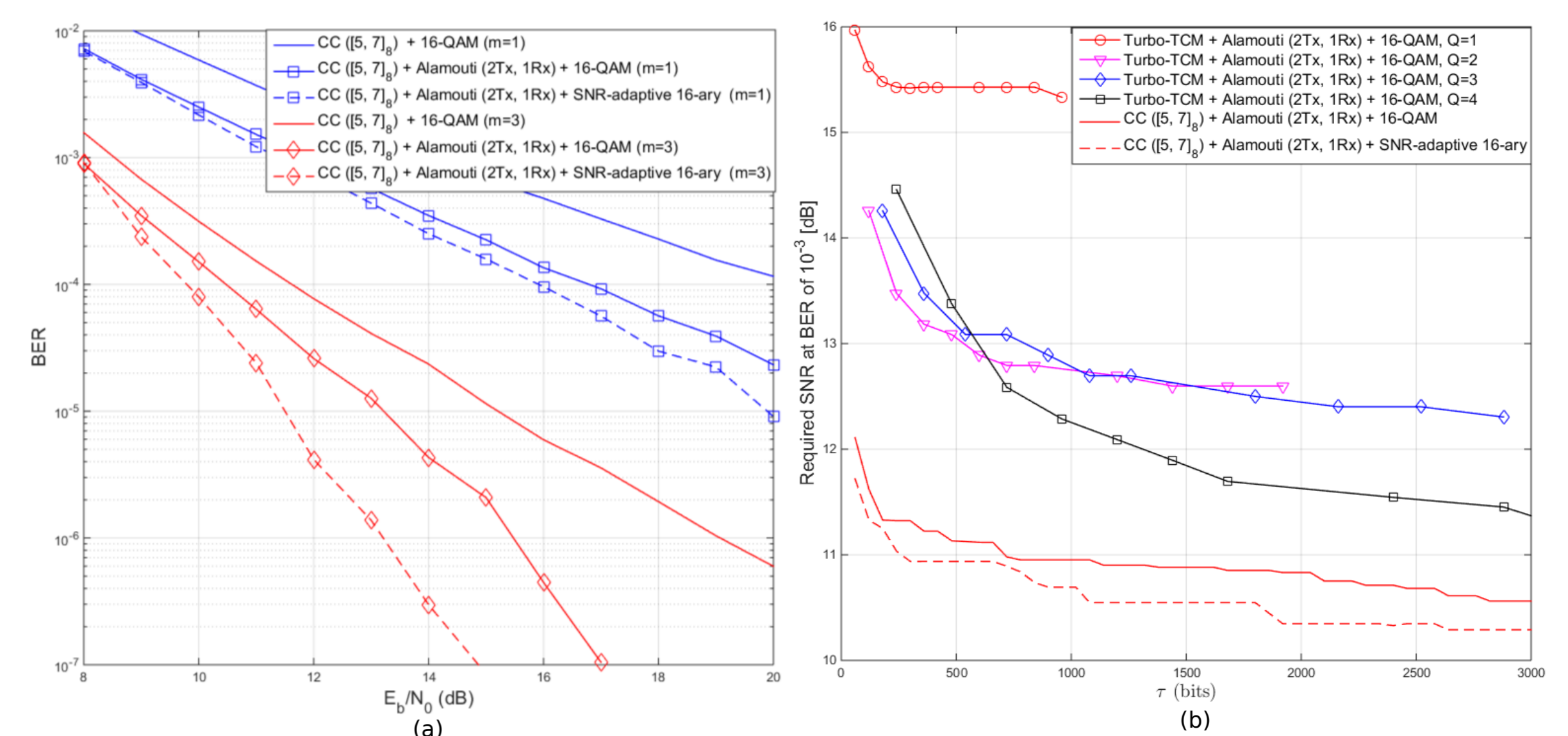

Figure 3: (a) The bit error rate comparison of the convolutionally encoded Alamouti coded SNR-adaptive 16-ary case and the convolutionally encoded Alamouti coded 16-QAM case for Nakagami-m channels, $m = 1$ and $m = 3$.
(b) The decoding latency comparison of the convolutionally encoded Alamouti coded SNR-adaptive 16-ary case, the convolutionally encoded Alamouti coded 16-QAM case, and the turbo-TCM encoded Alamouti coded 16-QAM cases with different iterative decoding numbers, Q= $\{1, 2, 3, 4\}$ over Rayleigh channels ($m = 1$).
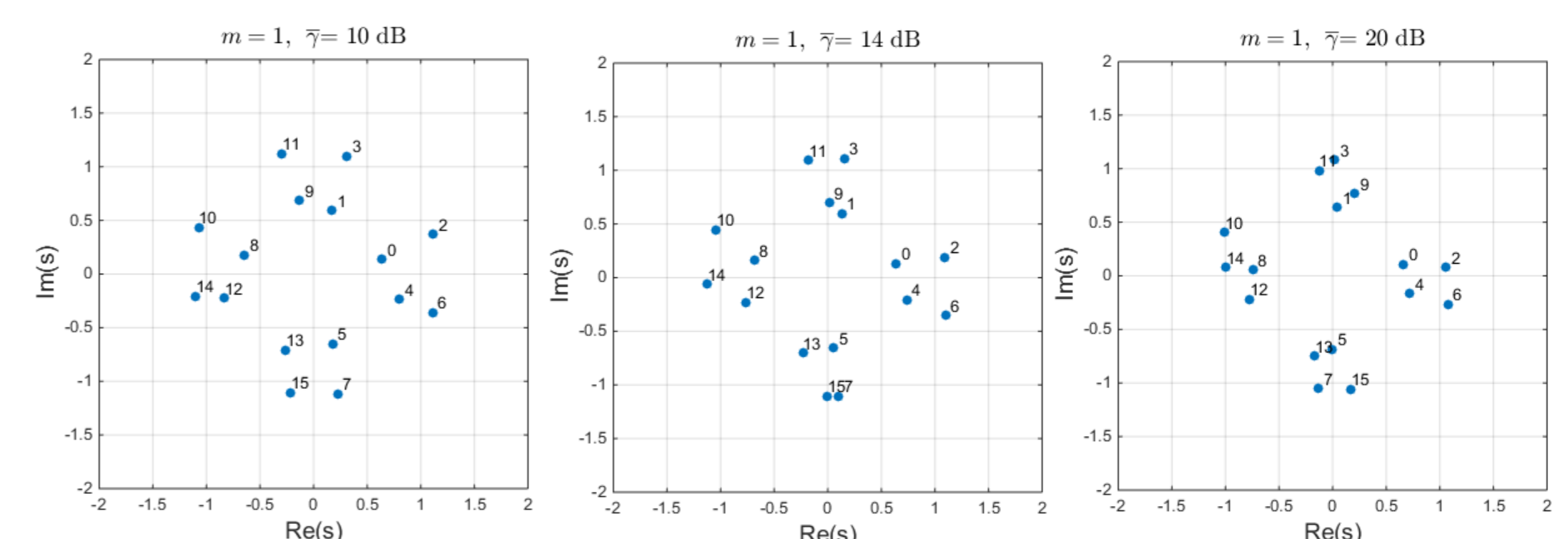

Figure 4: Optimized SNR-adaptive irregular constellations for convolutionally encoded $[5, 7]_8$ Alamouti coded scheme at $\bar{\gamma} = 10$ dB, $\bar{\gamma} = 14$ dB and $\bar{\gamma} = 20$ dB for Rayleigh fading channels ($m = 1$).

Conclusions

Non-iterative/one shot-decoding and the simplicity makes convolutional encoders attractive in particular 5G use cases where low-latency and low complexity are important attributes. Encouraged by this, SNR-adaptive convolutionally encoded OSTBC system is proposed where performance gain from constellation design is combined with diversity gain without any additional decoding complexity. From the simulations, it is observed that the notable gain in terms of error rates and decoding latency can be obtained from SNR-adaptive convolutionally encoded OSTBC system.

References

[1] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Select. Areas Commun.*, vol. 16, no. 8, pp. 1451–1458, 1998.

[2] M. C. Ilter, P. A. Dmochowski, and H. Yanikomeroglu, "Revisiting error analysis in convolutionally coded systems: The irregular constellation case," *IEEE Trans. Commun.*, vol. 66, no. 2, pp. 465–477, Feb. 2018.

[3] H. Ghazzai, E. Yaacoub, M.-S. Alouini, Z. Dawy, and A. Abu-Dayya, "Optimized LTE cell planning with varying spatial and temporal user densities," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1575–1589, 2016.

[4] S. Ebbesen, P. Kiwitz, and L. Guzzella, "A generic particle swarm optimization MATLAB function," in *Proc. American Control Conference (ACC)*, Jun. 2012, pp. 1519–1524.

[5] S. V. Maiya, D. J. Costello, and T. E. Fuja, "Low latency coding: convolutional codes vs. LDPC codes," *IEEE Trans. Commun.*, vol. 60, no. 5, pp. 1215–1225, May 2012.

[6] P. Robertson and T. Worz, "Bandwidth-efficient turbo trellis-coded modulation using punctured component codes," *IEEE J. Select. Areas Commun.*, vol. 16, no. 2, pp. 206–218, Feb. 1998.

Acknowledgements